THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 50, No. 4, April, 1971
Printed in U.S.A.

Statistical Circuit Design:

Linear Circuits and Statistical Design

By C. L. SEMMELMAN, E. D. WALSH and G. T. DARYANANI

(Manuscript received November 17, 1970)

The design of linear circuits requires the designer to consider element tolerances, distributions and correlations and how they interact with proposed manufacturing test limits and service conditions such as temperature and aging. Analytical methods can be used in a limited number of cases; simulation methods using large modern computers permit study of more complex design problems. The principal themes of this paper are the designer's needs for computational assistance and the ways in which computer programs may be organized to meet these needs. Two computer programs for this purpose are described. One is a general-purpose analysis program for large networks having any topology. The other is a more specialized program suitable for "biquad" networks which may be used as building blocks to form a variety of filter and equalizer networks.

I. INTRODUCTION

In the Bell System, the origin of interest in the statistical design of linear circuits is lost in history. It could have been no later than the earliest planning for repeaters and filters for carrier systems, as no such system is possible without precise control of frequency and loss characteristics. As the total installed cost of a carrier system is always much greater than the cost of the design effort, use of the most advanced design procedures and techniques known, including statistical methods, has always been justified.

The selection of tolerances for component parts is one of the many topics which is under the control of the designer. He must select values and specify manufacturing tests so as not to increase costs either by paying a premium for unnecessarily precise elements or by rejecting correctly assembled networks in production. In the past, the tools that have been available to the designer to assist in the determination of tolerance limits have ranged from primitive to highly sophis-

ticated and mathematical. In the present era, which is dominated by electronic computers, simulation methods may prove the most valuable of all. The purpose of this paper is to review some of the basic methods of calculating and combining deviation characteristics, pointing out the powers and limitations of these methods, and to describe additional capabilities obtainable from two general purpose computer programs. These enable the designer to determine the result, in production and in service, of specifying a given set of tolerances, distributions, correlations, tuning and testing procedures and service conditions. All the material is confined to linear networks, such as filters, equalizers and feedback amplifiers, which are designed and built to frequency domain specifications.

1.1 *Analytical Methods*

Engineers engaged in designing electrical networks and selecting tolerances for their components have traditionally used the classical design technique which consists of making a first approximation, computing or measuring its performance and refining the approximation. That this procedure worked well and converged rapidly when applied to network design can be seen from the following. The image parameter design process for filters and comparable methods for other networks gave the engineer an understanding of the sensitivity of the various components, i.e., which component has its greatest effect in what part of the frequency spectrum. This led intuitively to both the specification of the tests that would be sufficient to assure accurate manufacture and to the knowledge of which few components need careful scrutiny at any frequency, in order to select appropriate tolerances. A few calculations or laboratory measurements followed by some experimental arithmetic would then produce the necessary tolerance limits. As the element tolerances being used were small, the resulting network deviations were proportional and could be combined by simple addition. Adding the absolute values of the maximum permissible deviations would produce worst-case behavior and if system performance limits were met the job was done.

Mathematicians came to the assistance of the design engineers and devised clever labor-saving methods of determining the deviation effects and of combining them linearly. For example, the rate of change of the driving point impedance of a reciprocal network with respect to a branch impedance is given by

$$\frac{\partial Z}{\partial Z_x} = \left(\frac{I_x}{I}\right)^2$$

and the rate of change of a transfer impedance is

$$\frac{\partial Z_T}{\partial Z_x} = \left(\frac{I_x}{I_2}\right)\left(\frac{I_x'}{I_1'}\right)$$

where the symbols all represent complex quantities and are defined in Fig. 1.* One or two solutions for all the branch currents then permit each individual driving point or transfer impedance deviation to be calculated and scaled to represent the assumed element tolerances. As stated above, these scaled deviations may be added to evaluate the behavior of a network with any one set of element values.

More insight, however, may be obtained by drawing a polygon in the complex plane within which the impedance must lie. This may be done in the following manner: Select a plus or minus sign for the departure of each element from its nominal, as required to make the real component of each scaled deviation positive. Adding the scaled deviations with these signs will give the rightmost point of the polygon. Successive corners are obtained by reversing the signs of the scaled deviations, one at a time, in the sequence of their increasing angles. When all signs have been reversed, the leftmost point is obtained and a second reversal in the same sequence generates the other half of the convex polygon and returns to the first point.

Still more may be learned by superimposing a family of ellipses each of which contains a specified percent of the universe of networks which would be manufactured from elements having the assumed tolerances. This may be done if the distribution of each element value is gaussian and its standard deviation is known.[1,2]

Figure 2 illustrates both the impedance polygon and ellipses for a

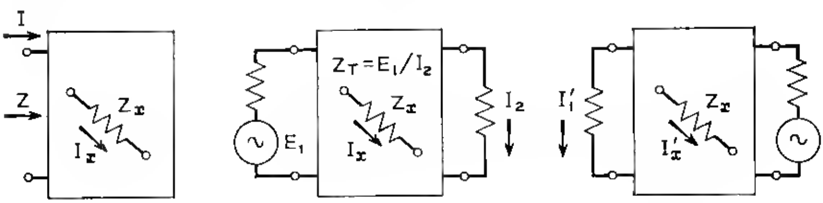

Fig. 1—Definition of symbols for driving point and transfer impedance deviations.

* The origin of these equations cannot be determined. They were in use in 1933 in the design of passive networks. They may be derived in a number of ways, including the use of network theorems and differentiating familiar expressions and identifying terms.

simple filter where the standard deviation is taken as 40 percent of the tolerance limit assumed for the polygon.

1.2 *Limitations to These Methods*

The first limitations are the facts that large tolerances do not permit linear extrapolation to large deviation shapes and that these do not combine linearly. Other difficulties arise when sophisticated design techniques such as filter synthesis by insertion loss design techniques[3] are used. The designer develops no "feel" for his design and the network performance becomes more sensitive to every element. Further, the elements may not have gaussian distributions. For example, 10 percent components may have a bimodal distribution resulting from a supplier's sorting the product and selling the 5 percent elements elsewhere. It may be impossible to determine whether high or low values of the elements should be used to evaluate worst-case behavior because a large change in one element's value may reverse the sign of the sensitivity for another element. Finally, the worst case may be so bad that system performance requirements are not met and tighter tolerances, factory adjustments, selective assembly, or reduced yield must be considered. Any of these increases the cost of manufacture, complicates the designer's task, and requires more powerful methods of determining the tolerances, adjustments and tests for production.

From the above discussion, it is apparent that only in rare cases will the selection of tolerances be a straightforward process where linear extrapolation and simple addition of deviation shapes will suffice. Frequently, the mathematical and engineering complexities put the designer's task beyond the limits of analytic methods available today.

II. MONTE CARLO TECHNIQUES

"Monte Carlo" computer techniques permit simulating the statistical behavior of random variables and imitating the tuning and testing actions performed in the factory. Such computer programs can take the place of a pilot production run, and furnish comparable information in a much shorter time and at a fraction of the cost.

In these methods the computer is programmed to generate a sequence of numbers which appear to a casual observer to be random, although their distribution and periodicities are known and carefully controlled. By mathematical transformations, these numbers are modified to represent possible values for each network element and to display the appropriate nominal value, tolerance, distribution shape and correlation with other elements. Further explanation of the

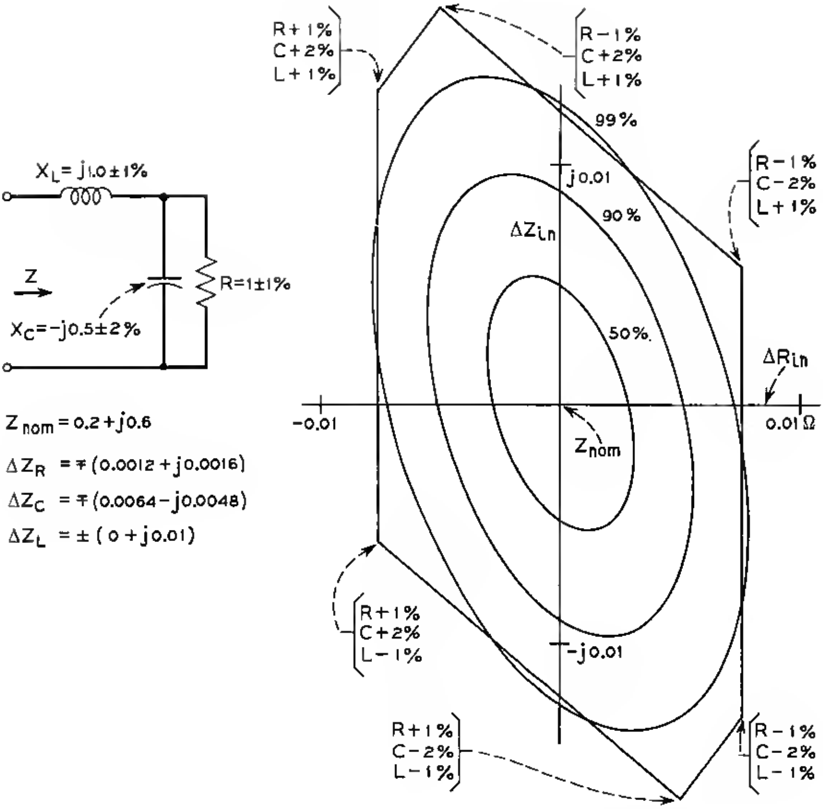

Fig. 2—Simple filter driving point impedance polygon and ellipses.

methods will be found in Appendix A. Many networks composed of such simulated elements are computed and the results compared with proposed test limits. A study of the computer output will enable the designer to determine the suitability of the nominal values, tolerances and tests.

The remainder of this article describes two computer programs, BELTAP and BITAP, which perform this simulation. The programs are addressed to different needs, but within its own area of application each is a general-purpose program.

BELTAP is a program for simulating and analyzing networks such as filters, equalizers, feed-back amplifiers, etc., having many elements and arbitrary topology. It computes insertion loss and return losses at both ends of the network. With it, the designer can obtain informa-

tion on test yields, the similarity between tests in this respect, element sensitivities, and selected scatter plots and histograms. This information should help him understand the complex interrelationships among element values, tolerances, distributions and tests which are not obvious from the network topology.

BITAP, on the other hand, can deal only with networks of a fixed topology, whose transfer function is restricted to a biquadratic function of frequency [defined in equation (1)]. By varying the element values used and connecting a number of such sections in tandem lowpass, highpass, bandpass filters and various types of equalizers can be constructed. BITAP facilitates specifying correlation among element values, factory tuning procedures, service conditions such as high or low temperature exposure and aging, and performance tests such as gain, phase and delay. If the service conditions are omitted, the tests become factory production tests; if included, end-of-life system performance checks.

III. DESCRIPTION OF BELTAP OPERATION

BELTAP is capable of performing tolerance analysis computations on a vast majority of the transmission networks in production today. The networks may consist of both lumped elements, having specified nominal values, tolerances and distributions, and fixed subnetworks, described by admittance matrices which are obtained either by calculation or measurement. BELTAP builds perturbed circuits by deviating each lumped element by a random amount within the specified tolerance and having a uniform, gaussian or arbitrary distribution shape. If correlation of deviations or tuning is required, special subroutines may be inserted. The perturbed circuits are then analyzed and insertion loss and input and output return losses are computed at each specified frequency. Special subroutines may be prepared to compute other quantities. For each perturbed network the complete set of elements and all measures of circuit performance are recorded on a disk file for later review by a postprocessor. The computed performance values are compared to the circuit requirements, to determine if the network passes or fails, and to calculate the production yields. The data on disk are then reviewed to collect other standard outputs and the listings, scatter plots and histograms requested by the designer. The output will be described more fully below. The variety of outputs available from BELTAP should be of great assistance to a designer who has little insight into the sensitivities of the elements, the interaction of their deviations and the relationship of both to the test limits.

3.1 *Specification of the Network*

A variety of types of information can be used to specify the network to be analyzed. It can contain resistors, inductors, capacitors and controlled current sources. Also, a quality factor, or Q, may be specified for the inductors and capacitors. Parallel and series resonant circuits may be specified by supplying only one of the reactive components and the resonant frequency. All the above items may be assigned tolerance limits and distribution shapes as required. Uniform and gaussian shapes are built-in; other special shapes may be read in at execution time. If correlations or factory tuning operations must be simulated, special subroutines can be written for this purpose.

BELTAP can allow fixed subnetworks to be included in the overall network. These subnetworks are specified by supplying their definite admittance matrices. The matrix data are generally obtained from an analysis of the subnetwork by a general-purpose linear network analysis program but may also be obtained from measurements.

The subnetwork feature allows BELTAP to perform a tolerance analysis on the critical components of a large network in a feasible amount of time. Consider, for example, the wideband amplifier shown in Fig. 3. The complete amplifier contains 80 nodes and 111 components. By blocking the circuit into two four-port subnetworks, as shown, the network to be computed is reduced to 24 nodes and 36 elements. The 36 elements are the critical components in the feedback paths and in sensitive positions in the forward path. The complete circuit contains

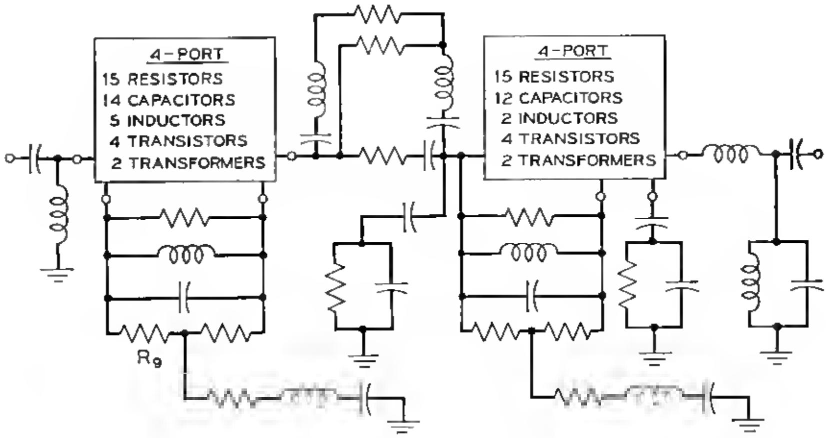

Fig. 3—Network for analysis by BELTAP, reduced to subnetworks.

four-port transformers, two-port transformers, and high-frequency transistors. Accurate lumped element models for these devices, particularly the transformers, do not exist. However, they can be easily incorporated into a BELTAP analysis by supplying their measured Y-matrix data (or, more likely, measured data reduced and transformed to give Y-matrix data).

3.2 BELTAP *Output*

A Monte Carlo analysis can produce such vast quantities of data that the designer, confronted with the complete results, might have difficulty identifying and extracting significant facts. It is the task of a postprocessor program to read the disk files and organize these data into brief comprehensible forms.

The postprocessor has two general classes of output, default output which is always produced and optional output which is produced only as specified. The default output is of a general nature and not of great detail or length. The requested output can be as voluminous as the engineer desires.

3.2.1 *Default Output*

The default output consists of four sections. The first section gives a summary of the test statistics. This includes the percent yield, average value, standard deviation, maximum value and minimum value for each test. It also gives the overall percent yield, combining all tests.

The second section of default output is a pseudo-correlation among pairs of tests. The pseudo-correlation is defined as the fraction of the networks for which the pass-fail result of the two tests agreed. For example, suppose ten networks were produced, the first seven passed test 1, the first nine passed test 2, and the remainder failed in both cases. The pseudo-correlation would then be 0.8 as networks 1 through 7 and 10 gave identical pass-fail results on the two tests. This output can be useful in reducing the number of tests performed without sacrificing reliability of performance. It may also be useful in determining an optimal sequence in which to perform the tests.

The third section of default output tabulates each parameter's name, nominal value, average value and standard deviation. This information is useful as a check on the random distributions and on the adequacy of the sample size. For example, the nominal value and the average value should be the same for all the parameters with a symmetrical density function.

The final section of default output gives the correlation among test

values and parameters. An approximation to the sensitivity of each test value to each parameter is also printed.

From these correlations and sensitivity estimates, the engineer is given insight into how individual circuit components affect his circuit's overall performance. More important, they give him a direction to follow in choosing those elements that may require tighter tolerances and those which may have broader tolerances.

3.2.2 *Optional Output*

The optional output has three sections: tabular output, scatter plots and histograms. Tabular output is merely a listing of the random element values and test values for each circuit. Circuits which failed tests are flagged for easy identification. Test results and element values may be intermixed so that the results of a test may be listed in a column adjacent to the values of an element to which the test is very sensitive.

There are three types of scatter plots available: test vs. test, element vs. test, and element vs. element. Figure 4 shows a scatter plot

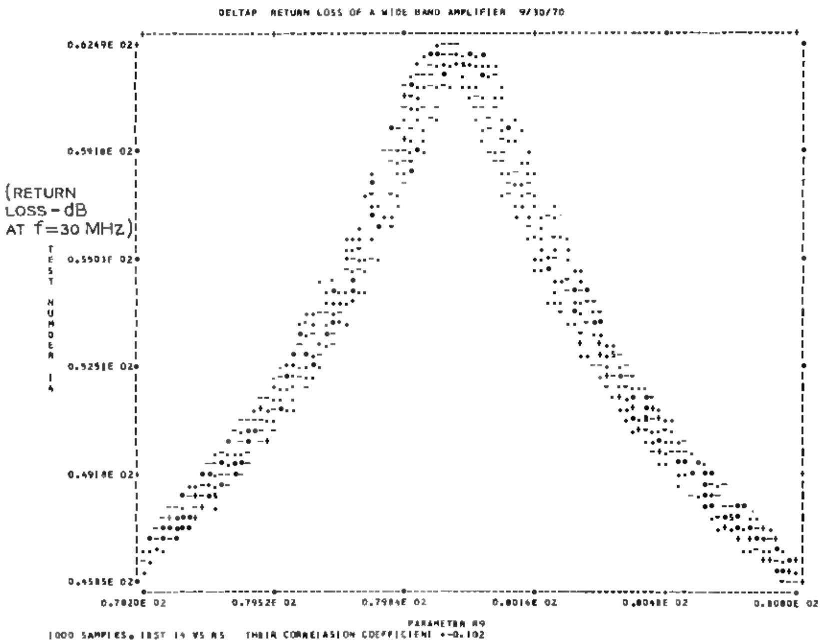

Fig. 4—Scatter plot of return loss vs. an element value.

of a test result vs. an element value. The various symbols printed on a scatter plot indicate the number of points that fell into a particular rectangular cell, the more points the blacker the symbol.

The scatter plots give an engineer more insight into how quantities in his circuit interact. They provide him with a type of pictorial information that is not available from correlation coefficients or sensitivity numbers. In Fig. 4, for example, the nominal value of the element occurs in the center of the plot, and at this point the slope of the test curve is near zero. Hence the partial derivative of the test result with respect to the element value and the correlation coefficient would be small. However, the test result is greatly influenced by the element value.

BELTAP can produce histograms of any number of element values and test values. Histograms of element values can be used as a check on the random number distributions and the density function routine in BELTAP. Histograms of the test values give the approximate shape of the probability density for the test.

3.3 *Summary of* BELTAP *Characteristics*

BELTAP provides a user with considerable flexibility in several respects. The network description language permits any configuration. The numbers of distribution types, branches, nodes and subnetworks are essentially unlimited as dynamic storage methods are used. As a result the size and number of networks computed will probably be restricted by computer running time, cost and numerical precision, rather than by array dimensions. Experience has indicated that when a very large network is encountered, the most sensitive elements can be selected by preliminary computer runs on portions of the entire structure. One or more additional runs can then be made with only these elements varying to obtain final data.

In addition to insertion loss and return loss calculations, which are provided by the program, users may write subroutines to evaluate other measures of network performance. Subroutines for simulating tuning adjustments, for introducing correlation effects, and for computing other measures of the network performance may also be written. The standard and optional outputs are designed to cover a broad range of types of data and methods of presentation. This variety should be helpful in discovering unsuspected interrelationships among tests and between tests and parameter values. For this reason, it is probably desirable to investigate considerably more tests than will actually be used in production.

IV. BITAP

The BITAP program has a very different purpose from that of BELNAP. BITAP can compute the behavior of only one kind of network, the "biquad." This network derives its name from the fact that its transfer function is a biquadratic function of frequency. It is an active network and has a fixed topology. The values of the resistors and capacitors used determine whether the biquad will be a lowpass, highpass, bandpass, loss equalizer or delay equalizer type of network. A number of biquad networks can be connected in tandem without interaction, so their gain and phase characteristics can be combined by simple addition and more complex structures realized. BITAP facilitates the investigation of realistic manufacturing conditions and environmental effects for biquad networks.

4.1 *Area of Application*

A common approach used in designing active filters is to build them with cascadable network sections. After the required transfer function has been determined, it is decomposed into a product of biquadratic functions of the form

$$\frac{V_{\text{out}}}{V_{\text{in}}} = \prod_{i=1}^{k} \frac{m_i s^2 + c_i s + d_i}{n_i s^2 + a_i s + b_i} \tag{1}$$

where m_i, c_i, d_i, n_i, a_i and b_i are real coefficients and $s = j\omega$.This decomposition may be done for any lumped, linear time-invariant network. The synthesis procedure consists of realizing each biquadratic function with an active network, which may be connected in tandem with other sections and will not interact with them. Examples of biquads are the state variable four amplifier biquad,[4] Moschytz's FENs[5] and the Sallen and Key networks.[6]

One of the aims in the realization of biquads is to make them isotopic, that is, to maintain the same network topology for all values of the coefficients m, c, d, n, a and b. If this can be done, the manufacturing process can be standardized, with obvious economic advantages. The state variable four amplifier biquad is completely isotopic; the FEN and Sallen and Key networks need a small number of topologies to realize all forms of the general biquadratic function. A circuit diagram of the four-amplifier biquad is shown in Fig. 5. By selecting the proper element values and making the right connections this network can be made to produce all the various filter functions—lowpass, high-

pass, bandpass, band-reject, notches and equalizers and delay shapes. The advantages of the realization justify writing the BITAP program.

We are concerned with the analysis of biquad networks under realistic manufacturing conditions and with environmental changes such as temperature, humidity and aging with time. Manufacturing conditions must include correlations, which will be present in thin-film and integrated circuit realizations and tuning adjustments needed to realize high-precision networks. Environmental factors must be taken into account as new manufacturing methods and materials are to be used and extensive experience with them is not available.

4.2 *Description of* BITAP

The basic approach is similar to that described above for BELTAP. For each element a nominal value, a tolerance and a distribution type are specified. The program generates random numbers which are transformed to become deviated element values. These are combined to form the six biquad coefficients, which then permit evaluation of the gain, phase or delay of each biquad at each frequency. These quantities are summed, the total is compared to the network requirements, and performance statistics are accumulated.

A number of additional considerations arise immediately. Because of the variety of types of biquad sections, a section type number is required to designate which set of formulas is to be used to calculate the biquad coefficients. The need to introduce correlation effects re-

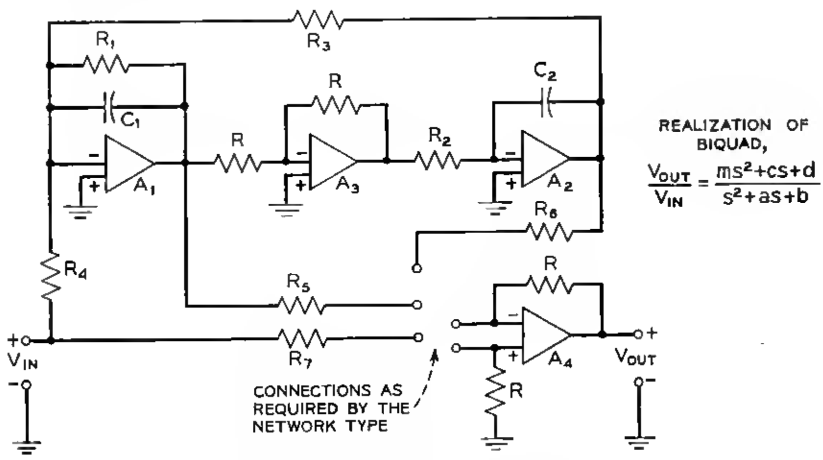

Fig. 5—Four-amplifier biquad circuit.

quires that every element value have an identifying serial number to permit cross-referencing. These serial numbers will also be useful in specifying tuning operations and in introducing environmental effects. Because of the limited number of section types, it has been possible to develop and include quite general methods of producing correlation, tuning, environmental effects and tests. More information on the methods of accomplishing these objectives will be given below. Where the standard methods are inadequate, new subroutines can be written and included in the program.

4.2.1 *Simulation of Manufacturing Conditions*

Due to the production process, parameters such as resistors, capacitors and amplifier gains have an actual value spread about the nominal value. This production tolerance on a parameter such as a resistor is represented by:

$$R_1 = R_0(1 + n_1) \tag{2}$$

where R_0 is the nominal value and n_1 is a random number whose range is the production tolerance on R_0. The distribution of n_1 could be gaussian, uniform, or any of several special shapes. The algorithm used to generate random numbers on the computers is discussed in Appendix A.

In integrated circuits, where all the elements are on the same chip, the values of the resistors track each other to some extent. Consider k resistors whose initial tolerances track. It has been found empirically that their values may be represented by:

$$\begin{aligned} R_1 &= R_{01}(1 + \rho n + (1 - \rho)n_1), \\ R_2 &= R_{02}(1 + \rho n + (1 - \rho)n_2), \\ &\vdots \\ R_k &= R_{0k}(1 + \rho n + (1 - \rho)n_k). \end{aligned} \tag{3}$$

The same random number n is used to perturb all k resistors. Each resistor $R_1, R_2, \cdots R_k$ is also perturbed by a separate random number $n_1, n_2, \cdots n_k$, respectively. ρ is a correlation factor which determines how closely the resistors track. The range of ρ is from 0 to 1. Complete correlation can be simulated by $\rho = 1$, and no correlation by $\rho = 0$. This correlation factor is different from the correlation coefficient used in statistics but has proven useful in the present application. The coding has been arranged so that the serial numbers mentioned earlier

effectively refer to both R and n values in equation (3), so that multiple correlations may be specified.

A practical problem in using the above sets of formulas, equations (2) and (3), is that of obtaining data on the distributions of the parameter values in production and on the correlation factor, ρ. The need for accurate statistical information may require that a major effort be made to obtain such data and keep it up-to-date. A companion paper, "Characterization and Modeling for Statistical Design," describes transistor characterization activities.

4.2.2 *Tuning*

The deviations from the nominal performance caused by the initial tolerances of components can be totally or partially corrected by tuning some of the network elements. The elements are tuned during the manufacture of the filter, at room temperature. Two types of tuning steps have been simulated on BITAP. In the first type, an element is adjusted so that a transmission requirement such as loss or phase is met at a given frequency. This type of tuning is used in adjusting for the bandwidth of a biquad section. In the second type of tuning, an element is adjusted for a peak (of gain or loss) at a given frequency. This is used in adjusting the resonant frequency of a pole or a zero. In both these tuning methods the variable element is changed continuously over a range. An additional subroutine could be written to simulate tuning in discrete steps, i.e., using a finite supply of element values close to the nominal, as this is an economical production method of adjusting the network performance. In production, the accuracy with which the element can be tuned depends on the sensitivity of the network to that element and on the precision of the measuring equipment. This is simulated by specifying a given nominal value for the tuning with a tolerance and a distribution.

4.2.3 *Simulation of Field Conditions*

The factors that caused the filter response to deviate from the nominal during manufacture were the initial tolerances of the components and inaccuracy in the tuning step. The environmental conditions that were assumed to exist during manufacture were room temperature and humidity. In the field, any change from these environmental conditions will cause the filter characteristics to change to some extent. The filter response will also change with time due to the chemical processes associated with aging. The temperature, aging and humidity effects in the field are modeled in the following way:

The value of a resistor (R_2) at T°C above room temperature is given by

$$R_2 = R_{02}(1 + \alpha T)$$

where R_{02} is the nominal value. The temperature coefficient (T.C.), α, is

$$\alpha = \alpha_0 + n_2$$

where α_0 is the nominal T.C. and n_2 the randomly distributed part of the T.C. The distribution function describing n_2 is usually gaussian.

The change in the value of a resistor or capacitor (R_3) with time is represented by

$$R_3 = R_{03}(1 + n_3) \tag{4}$$

where R_{03} is the nominal and n_3 simulates the random part.

The change due to humidity is represented by an equation similar to equation (4).

The combined effects of initial tolerance, temperature, aging and humidity is given by

$$R = R_0(1 + n_1)(1 + (\alpha_0 + n_2)T)(1 + n_3)(1 + n_4). \tag{5}$$

The random numbers associated with temperature coefficients (n_2), for components on an integrated circuit, are correlated. The same is true for the aging coefficient (n_3) and the humidity coefficients (n_4). These correlations are simulated just as in equation (3).

4.2.4 *Testing*

The values of the deviated parameters are used in the equations describing the biquad to evaluate the performance of the network. The biquad is analyzed several times using different sets of random numbers to simulate environmental and manufacturing conditions. The performance tests are spelled out in the form of filter function requirements (loss, phase or delay) at various frequencies. Three standard forms have been built into the program. These are:

(*i*) loss (phase, delay) at a frequency f_1 is between two limits, dB_1 and dB_2,
(*ii*) loss at a frequency f_1 is bounded below (or above) by dB_1, and
(*iii*) loss at a frequency f_1 is greater (less) than the measured loss at a reference frequency f_2 plus the amount dB_1.

More than one performance requirement may be specified at a frequency. It is expected that special subroutines will be written for

more sophisticated tests such as a resonant frequency, Q, and in-band ripple. It should be noted that if environmental conditions are included, the tests become service condition or end-of-life tests. If they are omitted, the tests may be interpreted as factory production tests.

4.2.5 *Output*

If, for a particular set of deviated elements, the network fails the required specifications, the complete network performance is stored. The output consists of a table which details the failed networks. If a test was failed, the performance function for that test is printed; otherwise, a blank is printed for the test.

The overall yield is computed as the ratio of the number of networks passing all tests to the total number of networks tested. Yields are also calculated for individual tests. The exact form of the output is illustrated by the example in the next section.

4.3 *Example*

The four amplifier biquad circuit of Fig. 5 will be used to illustrate the main features of BITAP. Its transfer function, in terms of the elements, is

$$\frac{V_{\text{out}}}{V_{\text{in}}} = \frac{+\frac{R}{R_7}\cdot S^2 + \frac{R}{R_7}\cdot\frac{1}{C_1}\left[\frac{1}{R_1} - \frac{R_7}{R_4 R_8}\right]\cdot S + \frac{R}{R_7 R_2 C_1 C_2}\left[\frac{1}{R_3} + \frac{R(R_5 + R_7) + R_5 R_7}{R_4 R_5 (R + R_6)}\right]}{S^2 + \frac{1}{R_1 C_1} S + \frac{1}{R_2 R_3 C_1 C_2}}.$$

The complete network contains two such sections connected in tandem as shown in Fig. 6, which also gives the element values. The resistors have a one percent manufacturing tolerance with a gaussian distribution. The capacitors are one percent elements with a flat distribution. The correlation factors, ρ, are 50 percent for both resistors and capacitors. The pole frequency of the first section is at 5 kHz, and R_3 is tuned to obtain this frequency within ± 1 Hz. This ± 1 Hz represents the accuracy of tuning and it is assumed to be distributed uniformly around the nominal 5 kHz. The bandwidth of the first section is tuned by R_1 to 100 Hz ± 1 Hz. Section 2 is not tuned. These initial tolerances and the tuning steps describe the manufacturing conditions. It is assumed that a BITAP run simulating the above conditions and manufacturing tests has already been made and 100 percent of the network passed. The filter performance is now to be tested at $+50°$C and at a time 20 years from manufacture. The T.C. of resistors is 130 ± 30 parts per million ($\rho = 80$ percent) and that of capacitors -135 ± 10

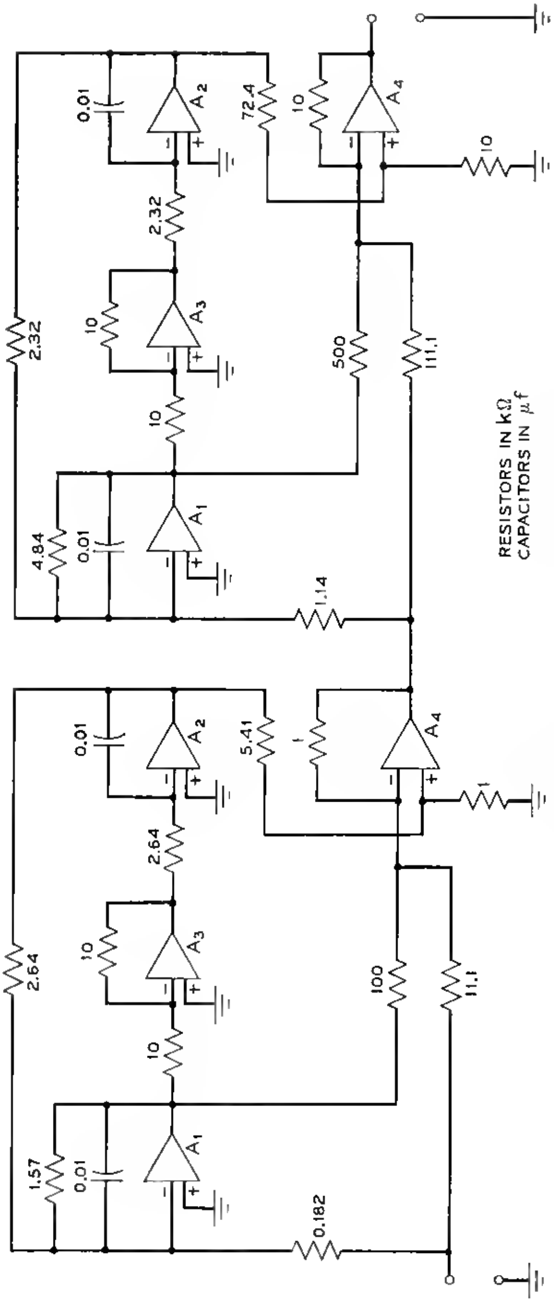

Fig. 6—Lowpass filter network for BITAP analysis.

ppm ($\rho = 50$ percent). The random parts (± 30 ppm and ± 10 ppm) are distributed normally about the nominal. The aging coefficient for the resistors and the capacitors in 20 years is ± 2 percent, uniformly distributed ($\rho = 50$ percent).

The nominal performance and the test requirements to be met at end-of-life are shown in Fig. 7. Figure 8 shows the results of a BITAP analysis of 100 filters and indicates that 82 percent will meet these service condition requirements. Figure 8 indicates both the frequencies where improvements are needed and the magnitude of the changes required. With the biquad type of realization and such data, the designer usually has no difficulty deciding which elements or tuning adjustments are responsible.

In order to improve the performance in the 1000- to 4000-Hz passband, the designer might examine several alternatives. A new nominal design with a smaller ripple or a better centered nominal loss would

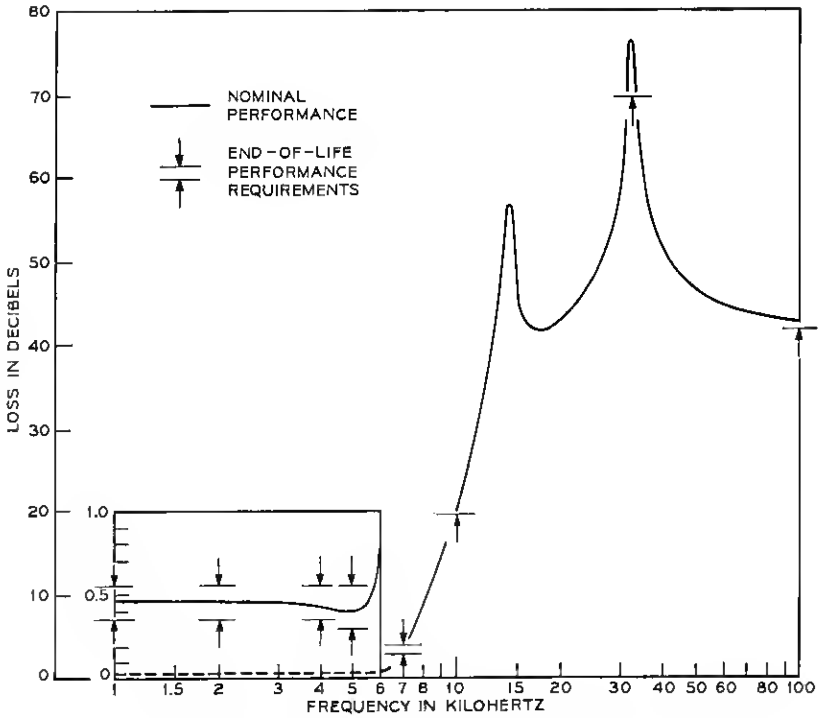

Fig. 7—Nominal performance and end-of-life requirement.

FREQ MEAS BELOW ABOVE NOM	1000.00 LOSS 0.5500 0.3500 0.4845	2000.00 LOSS 0.5500 0.3500 0.4738	4000.00 LOSS 0.5500 0.3500 0.4468	5000.00 LOSS 0.5500 0.3000 0.3947	7000.00 LOSS 4.000 3.000 3.391	10000.0 LOSS — 20.40 20.83	32000.0 LOSS — 70.00 76.70	100000. LOSS — 42.00 42.93
NET. NO.								
8	0.5569	0.5535	—	—	—	—	—	—
10	—	—	—	—	—	20.21	—	—
13	—	—	—	—	—	20.17	—	—
32	—	—	—	0.2998	—	20.32	—	—
36	—	—	—	—	—	20.39	—	—
45	—	—	—	0.2829	—	—	—	—
46	—	—	—	—	—	20.40	—	—
49	0.5506	—	—	—	—	—	—	—
50	—	—	—	—	—	20.21	—	—
51	—	—	—	—	—	20.16	—	—
57	—	—	0.3375	0.2819	—	—	—	—
60	—	—	—	0.2937	—	—	—	—
67	—	—	0.3364	0.2770	—	—	—	—
73	0.6009	0.6018	0.5891	—	—	—	—	—
76	0.5584	0.5641	0.5606	—	—	—	—	—
78	0.5657	0.5544	—	—	—	—	—	—
85	—	—	—	—	2.961	20.13	—	—
93	0.5620	0.5637	0.5564	—	—	—	—	—
NO. FAIL	6	5	5	5	1	8	0	0

FINAL YIELD = 82.000 PERCENT

Fig. 8—BITAP yield analysis.

entail no added cost. Use of closer tolerances for some components or more precise tuning might also provide sufficient improvement, but at some added expense. If none of these works, addition of a third section should be evaluated.

4.4 *Appraisal of* BITAP

BITAP is capable of analyzing up to 20 biquad sections. The number of performance tests is limited to 30. Ten types of biquad sections are currently included in BITAP. The addition of more biquad sections merely involves describing the biquad coefficients (m, c, d, n, a, b) in terms of the elements of the biquad and providing equations for the tuning steps. The rest of the program is identical for all biquads.

BITAP is an exact analysis package and as such it can be used in other steps of the design process. For instance, it could be used to determine whether or not a filter needs to be tuned, and to decide on the number of tuning steps needed. BITAP could also be used in determining the best (cheapest) components that could be used in a circuit while meeting all requirements.[7]

It has been shown how BITAP can be used to evaluate the performance of networks of the biquad family at the time of manufacture and under (end-of-life) field conditions. The information obtained here can be used to decide on the tests to specify during manufacture. The analysis under end-of-life conditions gives a measure of the adequacy of the design under extreme conditions.

V. CONCLUSIONS

We have described some simple analytic methods of calculating and combining deviation shapes and have discussed the areas in which they are useful and the respects in which they are limited. We have also presented two simulation procedures, which rely on the speed and large storage capacity of modern digital computers. Each is a general-purpose program in that it can be used to analyze statistically a large number of networks of an appropriate type. Although the BITAP networks form a subset of those analyzable by BELTAP, the two programs provide entirely different facilities for their users. BELTAP provides complete freedom in the network topology; BITAP uses a functional input form, which reduces computing time. BELTAP facilitates exploration of concealed interactions among component values and test results; in BITAP it is assumed that the few simple network functions allowed are completely understood in this respect. BELTAP provides no built-in cor-

relation, tuning, environmental exposure, or phase or delay evaluation but does include return losses; BITAP includes all of these except return losses which are of no interest.

Under certain conditions, the least expensive manufacturing method may require discarding some of the product. The simulation programs should be capable of demonstrating this, if it is true. They may also provide information that is useful in correcting troubles in the manufactured product. Their usefulness will, of course, depend on the availability of accurate statistical information.

Clearly, useful as they are, neither of these programs can be termed completely general purpose. Much work remains to be done in both areas of application. It is considered that the two programs make a very worthwhile contribution to a designer's repertoire and may point the way for still more comprehensive tools in the future.

VI. ACKNOWLEDGMENTS

The authors wish to express their appreciation to F. P. Sansone and Mrs. H. D. Reinecke for their work leading to BITAP.

APPENDIX A

Both BELTAP and BITAP use a congruential random number generator in which the modulus is an integral power of a prime number and the multiplier is a primitive root.[8] This method permits generating a sequence of numbers which has a flat distribution and a period which is made just large enough to produce the number of random numbers required. The modulus, M, is obtained by raising three to a power, n, such that the period, $2 \cdot 3^{n-1}$, exceeds the required value. The multiplier used is $2 + 9 \cdot k$, where k is an integer chosen to make the multiplier approximately equal to the square root of the modulus. It was found empirically that this function of k produces a primitive root, that is, a multiplier having the period given above, which is the longest that can be obtained for the assumed modulus.

A simple example using small numbers will illustrate the process. Suppose five networks are to be calculated and each contains three elements with distributions. Fifteen random numbers will be required. A value of three for n will produce a sequence of $2 \cdot 3^2 = 18$ numbers before repetition begins. The modulus M is $3^3 = 27$ and multipliers of 2, 11 or 20 are possible. Suppose the sequence begins with 1 and a multiplier of 11 is used. The second number is $1 \cdot 11 = 11$. The third is obtained by multiplying the second by 11, dividing by 27 and using

the remainder, 13. The fourth is 13 times 11 modulo 27 = 8, etc. The complete sequence is: 1, 11, 13, 8, 7, 23, 10, 2, 22, 26, 16, 14, 19, 20, 4, 17, 25, 5, 1. It should be noted that the nineteenth number, 1, is a repetition of the first and that all numbers less than 27 are present except multiples of three. The first three numbers will be used for the three random elements in the first network, the next three for the second network, etc.

The random numbers are shaped to become element values having the desired tolerance and distribution in the following manner. The first transformation is to put them into a range from -1 to $+1$ by dividing by half the modulus and subtracting 1.

$$n_b = \frac{n_a \cdot 2}{M} - 1. \tag{6}$$

The smallest random number, 1, yields $-0.925925 \cdots$, while the largest gives $+0.925925 \cdots$. With a larger modulus the numbers will approach -1 and $+1$ more closely. If an element has a flat distribution, the random value may be produced as follows:

$$R = R_0(1 + t \cdot n_b/100) \tag{7}$$

where R is the random value, R_0 is the nominal value for that element, and t is its (symmetrical) tolerance in percent. If the distribution is not flat, the values of n_b are transformed as shown in Fig. 9. For each value of n_b the proper line segment is identified and a value of n_c is calculated by interpolation. n_c is then used in place of n_b in equation

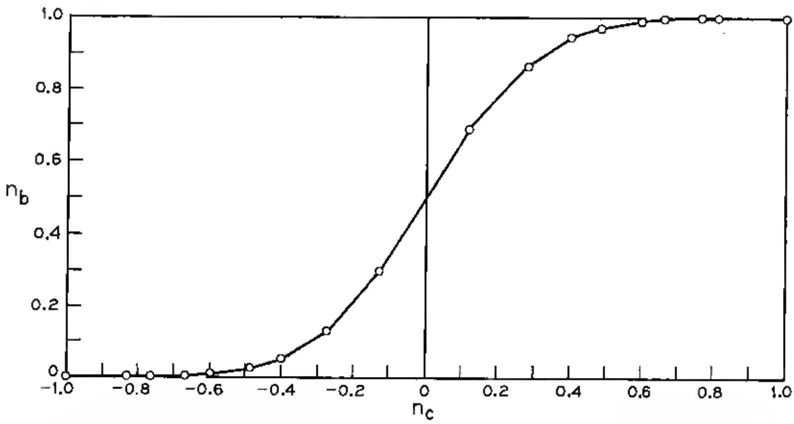

Fig. 9—Cumulative gaussian probability distribution approximated by straight-line segments.

(7) to calculate R. The coordinates for the gaussian distribution are permanently stored in the program. Other distributions can be used by reading in the appropriate sets of breakpoints.

If correlation is required, values of n_b and/or n_c may be combined as follows:

$$n_d = \rho n_x + (1 - \rho) n_y . \tag{8}$$

Here ρ is a correlation factor and n_d, n_x and n_y all represent random numbers in the -1 to $+1$ range, associated with selected elements, parameters, field conditions, and/or tuning tolerances. It is to be noted that n_d may be used to evaluate an R, using equation (7), or it may be used as an n_x or n_y in additional equations of the same form as (8). If the latter is done, the result will be partially correlated to three other parameters.

When equation (8) is used to introduce partial correlation and n_x and n_y have average values of zero and are uncorrelated, it can be shown that the variance of n_d is given by

$$\sigma_d^2 = \rho^2 \sigma_x^2 + (1 - \rho)^2 \sigma_y^2 . \tag{9}$$

It may also be shown that the tracking coefficient between n_d and n_x is given by

$$r = \sum_1^N n_d n_x / N \sigma_d \sigma_x = \rho \sigma_x / \sigma_d . \tag{10}$$

The tracking between two different distributions n_d formed using the same n_x and different n_y is discussed in the following paper.

REFERENCES

1. Hoyt, R. S., "Probability Theory and Telephone Transmission Engineering," B.S.T.J., *12*, No. 1 (January 1933), pp. 35–75.
2. Boghosian, W. H., unpublished memorandum, May 3, 1938.
3. Darlington, S., "Synthesis of Reactance 4-Poles," J. Math. and Phys., XVIII, (September 1939), pp. 257–353.
4. Tow, J., "A Step-by-Step Active Filter Design," Spectrum, *6*, No. 12 (December 1969), pp. 64–68.
5. Moschytz, G. S., "Active RC Filter Building Blocks using Frequency Emphasizing Networks," IEEE J. Solid State Circuits, *SC-2*, No. 2 (June 1967), pp. 59–62.
6. Sallen, R. P., and Key, E. L., "A Practical Method of Designing RC Active Filters," I.R.E. Trans. Circuit Theory, *CT-2* (March 1965), pp. 74–85.
7. Karafin, B. J., "The Optimum Assignment of Component Tolerances for Electrical Networks," B.S.T.J., this issue, pp. 1225–1242.
8. Hutchison, D. W., "A New Uniform Pseudo-Random Number Generator," N66-32469, Stat. Dept., U. of Calif., Berkeley, April 27, 1965.